

ACADGILD ASSIGNMENT 2.2

1. Read multiple JSON files into a directory to convert into a dataset. I have files text1, text2, text3 in the directory JSON.

Answer:

Read the JSON File

The JSON file is read by R using the function from **JSON()**. It is stored as a list in R.

```
# Load the package required to read JSON files.
```

```
library("rjson")
```

```
# Give the input file name to the function.
```

```
result <- fromJSON(file = "input.json")
```

```
# Print the result.
```

```
print(result)
```

When we execute the above code, it produces the following result –

```
$ID
```

```
[1] "1" "2" "3" "4" "5" "6" "7" "8"
```

```
$Name
```

```
[1] "Rick" "Dan" "Michelle" "Ryan" "Gary" "Nina"  
"Simon" "Guru"
```

```
$Salary
```

```
[1] "623.3" "515.2" "611" "729" "843.25" "578" "632.8" "722.5"
```

```
$StartDate
```

```
[1] "1/1/2012" "9/23/2013" "11/15/2014" "5/11/2014" "3/27/2015"  
"5/21/2013"  
"7/30/2013" "6/17/2014"
```

```
$Dept
```

```
[1] "IT" "Operations" "IT" "HR" "Finance" "IT" "Operations"  
"Finance"
```

Convert JSON to a Data Frame

We can convert the extracted data above to a R data frame for further analysis using the **as.data.frame()** function

```
# Load the package required to read JSON files.
```

```
library("rjson")
```

```
# Give the input file name to the function.
```

```
result <- fromJSON(file = "input.json")
```

```
# Convert JSON file to a data frame.
```

```
json_data_frame <- as.data.frame(result)
```

```
print(json_data_frame)
```

When we execute the above code, it produces the following result

	id,	name,	salary,	start_date,	dept
1	1	Rick	623.30	2012-01-01	IT
2	2	Dan	515.20	2013-09-23	Operations
3	3	Michelle	611.00	2014-11-15	IT
4	4	Ryan	729.00	2014-05-11	HR

5	NA	Gary	843.25	2015-03-27	Finance
6	6	Nina	578.00	2013-05-21	IT
7	7	Simon	632.80	2013-07-30	Operations
8	8	Guru	722.50	2014-06-17	Finance

```

1 install.packages("rjson")
2 # Load the package required to read JSON files.
3 library("rjson")
4 Give the input file name to the function.
5 # Give the input file name to the function.
6 result <- fromJSON(file = "input.json")
7 # Print the result.
8 print(result)
9 # Convert JSON file to a data frame.
10 json_data_frame <- as.data.frame(result)
11 print(json_data_frame)

```

```

C:/Users/10461882/Desktop/
result <- fromJSON(file = "input.json")
> # Give the input file name to the function.
> result <- fromJSON(file = "input.json")
> # Print the result.
> print(result)
$ID
[1] "1" "2" "3" "4" "5" "6" "7" "8"

$Name
[1] "Rick" "Dan" "Michelle" "Ryan" "Gary" "Nina" "Simon"
[8] "Guru"

$Salary
[1] "623.3" "515.2" "611" "729" "843.25" "578" "632.8" "722.5"

$StartDate
[1] "1/1/2012" "9/23/2013" "11/15/2014" "5/11/2014" "3/27/2015" "5/21/2013"
[7] "7/30/2013" "6/17/2014"

$Dept
[1] "IT" "Operations" "IT" "HR" "Finance" "IT"
[7] "Operations" "Finance"

> # Convert JSON file to a data frame.
> json_data_frame <- as.data.frame(result)
>
> print(json_data_frame)
  ID Name Salary StartDate Dept
1  1 Rick  623.3  1/1/2012   IT
2  2  Dan  515.2  9/23/2013 Operations
3  3 Michelle 611 11/15/2014   IT
4  4  Ryan  729  5/11/2014   HR
5  5  Gary 843.25 3/27/2015 Finance
6  6  Nina  578  5/21/2013   IT
7  7 Simon 632.8 7/30/2013 Operations
8  8  Guru 722.5 6/17/2014  Finance

```

2. Parse the following JSON into a data frame.

```
js<-'{"name": null, "release_date_local": null, "title": "3 (2011)",
```

```
"opening_weekend_take": 1234, "year": 2011,
"release_date_wide": "2011-09-16", "gross": 59954
}
```

```
> js<-'[{"name": null, "release_date_local": null, "title": "3 (2011)",
+ "opening_weekend_take": 1234, "year": 2011,
+ "release_date_wide": "2011-09-16", "gross": 59954}]'
>
> js <- fromJSON(js)
> js <- lapply(js, function(x) {
+   x[sapply(x, is.null)] <- NA
+   unlist(x)
+ })
> do.call("rbind", js)
      name release_date_local title      opening_weekend_take year
[1,] NA      NA              "3 (2011)" "1234"                "2011"
      release_date_wide gross
[1,] "2011-09-16"      "59954"
> |
```

3. Write a script for Variable Binning using R.

```
31 V<-1:4000
32 printV
33 print(v)
34 V<-seq(1:100)
35 print(v)
36 tapply(v,cut(v,60))
37
```

```
> tapply(v,cut(v,60))
Error: unexpected '>' in ">"
> tapply(v,cut(v,60))
[1] 1 1 2 2 3 4 4 5 5 6 7 7 8 8 9 10 10 11 11 12 13 13 14 14 15 16 16
[28] 17 17 18 19 19 20 20 21 22 22 23 24 24 25 25 26 27 27 28 28 29 30 30 31 31 32 33
[55] 33 34 34 35 36 36 37 37 38 39 39 40 40 41 42 42 43 44 44 45 45 46 47 47 48 48 49
[82] 50 50 51 51 52 53 53 54 54 55 56 56 57 57 58 59 59 60 60
```